MASTER IN BUSINESS ENGINEERING PROFESSIONAL FOCUS IN DATA SCIENCE

On the Conquest of Scholarly Data
What Are the Key Drivers of Successful Scholars ? - A Machine Learning Approach

Selmani, Leutrim

Award date:
2021

Awarding institution:
University of Namur

On the Conquest of Scholarly Data: What Are the Key Drivers of Successful Scholars? – A Machine Learning Approach

Leutrim SELMANI

Directeur: Prof. S. Bouraga

Mémoire présenté
en vue de l'obtention du titre de
Master 120 en ingénieur de gestion, à finalité spécialisée
en Data Science

ANNEE ACADEMIQUE 2020-2021

Université de Namur, ASBL
Faculté des Sciences économiques, sociales et de gestion – Département des Sciences de gestion

Rempart de la Vierge 8, B-5000 Namur, Belgique, Tel. +32 [0]81 72 49 58/48 41

Contents

Acknowledgment

First of all, I would like to thank my promoter whose suggestions and availability were of great help to me.

I would also like to thank all the professors that were in charge of my formation and that helped me throughout this long journey. This works concludes all the energy and the efforts I provided during these last years.

Finally, I would like to thank my family and my brothers that have always been there for me, without whom I would not be there today and, of course, al hamdulillah.

Chapter 1

Introduction

In recent years, the appearance of data and its exponential increase have been a key factor in the development of a lot of different fields. An increasing amount of papers are unanimous on the fact that data has become essential. This is why [43] states that *"The digital world is facing the aftermath of data explosion"*.

Data has been present in several forms, may it be numbers, words,... and terms like big data or data deluge [43] have emerged to describe the huge amount of data that is breaking through in all fields.

At the same time, there has been an increase of scientific papers and, as a direct consequence, an expansion of the scholarly data. Indeed, scholarly information has been rapidly growing because of the current capabilities to store and to tag all the research works across academia and industry [91] produced in the last decades. In [85], they say that *"The term Big Scholarly Data is coined for the rapidly growing scholarly data"*, and also shows to us how scholarly data is being increasingly matched with the term big data.

In today's world, the number of people that have chosen a life of scholarship, the number of researchers as well as educators keeps on increasing [1]. It is mainly due to the fact that they want to have an impact through having a positive effect on students, practitioners, colleagues and even society [1]. All things considered, both the increase of scholarly data and practitioners, offer an open reflection: how to value this overwhelming amount of scholarly data? How can we learn from the past data to influence the future scholarly? How to better grasp the concepts and theories that govern this discipline? Etc.

These are the main reasons why the *"scholarly world"* has already been studied through various perspectives. Some works like [90] focused on the scholar on a more general basis. For instance, authors of [90] have presented a technique to classify given scholars based on their topics.

Other works, like our current one, focused on the *"impact of the scholar"*. Indeed, we can wonder how to define the impact of a scholar (and its authors) and which are the elements that are the most likely to increase/decrease this *"impact"*. This raises a lot of questions like those of [2]: *"Who are the scholars with the greatest impact [...] ? What is the relative impact of individual articles, as well as entire journals ?"*. In the end, which factor is going to show that there was indeed a positive effect? To explore all the issues related with the concept of *"scholar impact"*, we can find a lot of scientific work that have been conducted in the literature. First, some works tried to clarify the concept of *"impact"* in a scholarly context. This is the case of [71] which revises all the metrics that are related to the concept of scholar impact and that are used by the current libraries. The objective of the authors is to clarify these metrics and to assist practitioners with the use of these metrics.

[2] tries to determine how the scholar's impact may be influenced by the impact of *"external stakeholders"* (those who are not *"academic"*) and *"internal stakeholders"* (those

who are *"academic"*).

Some works focus on specific types of scholars. This is the case of [44] whose authors work on what they called *"dissertations"*, i.e. the most frequent productions of young researchers. Dissertations are often excluded from traditional indexes making their impact hard to assess. To counter this, [44] proposes a new technique to assess dissertation's impact.

Other works, on their side, focus on how the scholar impact definition varies from one specific topic/discipline to another. One example is [60] that especially refines some impact metrics in the field of accountability. Through the use and analysis of well-known impact metrics, the g-index and the h-index, authors of [60] propose a more robust approach of scholar impact in their field of interest. Another example is [31] that concentrates on scholar collaboration (between authors). Their objective is to see the influence of the scholar collaboration, first, through different topics and, second, on the scholar impact and quality.

On another side, in [50], they analyse the influence of browsers and databases like Scopus and Google Scholar on the impact of scholars compared to more traditional databases.

While we can find a lot of work on the "scholar impact", only few of them focus on determining the key drivers of the "scholar impact". Indeed, we can intuitively think of factors that may influence the impact of a scholar. But how can we analyse them more formally? Some works like [53] attempt to sketch out an answer to this issue. The authors try to build up a recommendation system that would predetermine that a scholar will become what they call *"Academic Rising Stars"*.

Given that, and by also introducing Machine Learning in the field of big data, there is a lot of research to make and manipulations to perform with the available scholarly data to extend and enhance current literature. Machine learning will allow pattern recognition, classification and prediction [27] of this scholarly data, which, in return, will allow us to understand this data from a different perspective. The goal is to better understand the key drivers of the *"scholar impact"*. By which factor is it the most influenced? Why and how? The underlying idea is to better grasp what influences the impact of scholars. The benefit would be twofold; first to gain knowledge in the context of scholar by valuing the huge amount of data that is at our disposal, second it would provide future authors with recommendations about what needs to be well adjusted when producing a scholar according to their objectives.

In this paper, we will be trying to find tendencies from the large amount of data we get from scholarly data. The ambition is to run Machine Learning models with the scholarly data in order to train models to predict the impact of the paper. We hope that it will ultimately give insights on which features influence this scholar impact.

To proceed, the remainder of the work is structured as follows: the environment is exposed in section 1. Then, we recall the theoretical background in section 2. All the scholarly data terms and the machine learning theories are explained. In section 3, some related works are presented to assess the literature regarding the scholar context. The methodology is presented in section 4. We present first the data sources, second the data pre-processing, and third our implementation strategy regarding the machine learning models we run. We highlight the results of the work in section 5. We conclude the paper in section 6 with some limitations and future works.

Chapter 2

Theoretical Background

As mentioned, the purpose of this research is to analyze scholarly data and to find what are the parameters within it that have the highest impact in the academic/scientific community. To achieve this, we will first be defining scholarly data and explaining the different terms related to it. After that, we will explain what Machine Learning is, and we will be showing the different methods of using it. Hence, by explaining those concepts, we will be able to properly analyze and extract what is relevant for us in the scholarly data.

2.1 Scholarly Data

2.1.1 Definition

In order to understand what is going to be analyzed, we need to give a better view of what Scholarly Data is. Also, we will need to explain the related vocabulary as well as the Scholar Impact, and this is what the subsequent paragraphs are about.

The starting point of the scholarly data is a scientific production written by some experts, like a conference paper, a journal paper, a scientific report, a dissertation, etc. The latter comes with a considerable amount of metadata about the production itself, its publication, its authors, etc. They all make the scholarly data. To be concise, we will use the term *"article"* to refer to the *"scientific production"*.

More formally, [47] describes the scholarly data as:

"Scholarly data contain abundant academic resources such as scholarly documents (i.e., papers, books, patents, and scientific reports) as well as associated data including information of authors, citations, figures, tables, etc."

Therefore, the information contained in Scholarly datasets is going to be heterogeneous with various entities [47]. Figure 2.1 gives a clear structure of the information that is likely to be encountered in such datasets.

By starting from the *"Papers"*, we can see in Figure 2.1 that a paper is written by authors and/or co-authors and is published at a conference or a journal. This means that a paper contains a series of bibliographic data that are [13]:

- its authors/co-authors,
- A title,
- A publisher,
- Its publication information,

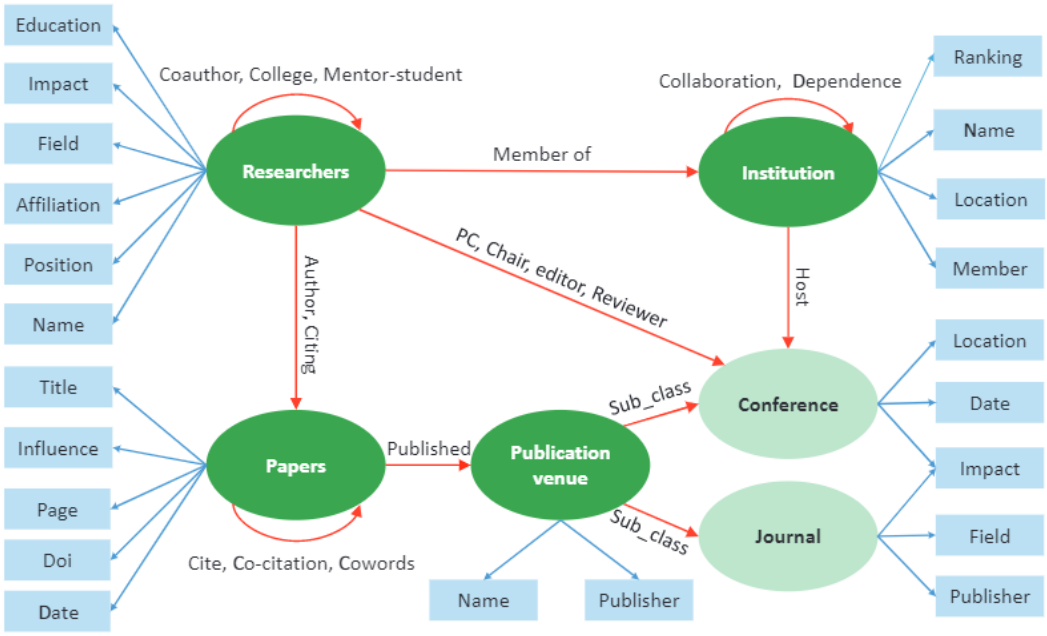


Figure 2.1: Scholarly Data structure extracted from [85]

- Its identifiers, such as the DOI.

In addition to that, there is also information such as the authors' contact information and his corresponding affiliation, i.e. their research centre or competent institution under which they are publishing, the date of publication of the article and the references of the article that redirect to other articles that also contain this data [13, 14, 10].

Apart from that, the *"Researchers"* also have several interesting pieces of information, such as an education, an impact, an affiliation,... that we can see in 2.1. Also, data can be collected from the articles that they have written or for which they have collaborated. An author often writes through a qualified institution from which we can also collect information and the publication of an article is mainly made through either a journal or a conference, but both provide different kinds of information, except for the impact.

In the end, where can we find such data ? This kind of scholarly data is available through open access or through libraries, and can be directly accessed with an adapted browser or exported in specific formats such as json, csv, xlsx,... [78, 77] to then be manipulated with a programming language or accessed through an API [24]. Scholarly datasets can also be found in papers on the topic. For instance, [85] proposes several data sources with Figure 2.2.

Data Set	Discipline	Size	Updated time	Downloading Link
Aminer	Computer Science	710MB	$2013-02-26$	https://aminer.org/billboard/AMinerNetwork
APS	Physics	1.21GB	$2014-07-21$	http://journals.aps.org/datasets
DBLP	Computer Science	297MB	$2015-09-05$	http://dblp.uni-trier.de/xml/
MAG	Multidisciplinary	29.8GB	$2015-08-31$	http://research.microsoft.com/en-us/projects/mag/

Figure 2.2: Examples of well-known Scholarly datasets extracted from [85]

2.1.2 Scholarly Impact

The growing use of impact metrics for the evaluation of scholars, institutions, journals,... has proven that there is an essential need for *"means to compare scientific impact"* [42]. Intuitively, the impact of the scholar should refer to the kind of disruption it leads to within its field of interest when published, read or used; how much does it bring to the field? How much success does it get? – But how to define *"success"* in this context? – How much credibility does it receive and by who? By which readers with which level of impact? This impact, of course, will be influenced by external factors, like the current fame of the field, but also by internal factors, like the impact of the authors themselves. It all shows the complexity of the notion of impact. The latter will somehow represent multidimensional notions. While its definition, and computation, seem highly complex, its core objective and utility do not remain less important. Indeed, the impact of a scholarly work partly determines the way resources are allocated and the way individuals and departments are rewarded [2]. As a consequence, numerous articles that discuss the importance of this topic and that try to tackle the issue have been proposed.

Scholarly impact has been given different definitions across the literature. However, the definition and the assessment of the impact and the value of a scholarly work remains challenging [1]. But this step is mandatory in order to be able to go further. In [1], we can learn that:

"Until there is clarity on how scholarly impact is defined and assessed, calls to be 'impactful' (akin to calls to behave in any other way) are unlikely to be effective in inspiring actions needed to achieve this result."

To recall what was mentioned in [1], this impact will allow us to know if a given research, work or paper is going to have a positive effect on interested parties.

As said in [2], *"Scholarly impact [...] has traditionally been equated with number of citations – be it for individuals, articles, departments, universities, journals or entire fields"*. Hence, scholarly impact is yet to be defined according to what is being researched, but it will allow us to answer several questions, such as those stated in [2]: *"Who are the scholars with the greatest impact [...] ? What is the relative impact of individual articles, as well as entire journals ?"* and many more.

In the measure of the impact of articles for example, taking into account the number of citations is used to define scholarly impact from a researcher perspective. The resulting number of citations will help to compare published works, researchers as well as fields between themselves [1].

Furthermore, research contributions from different scientific and technological fields have brought new tools and techniques that allow us to assess scholarly impact in several new ways [88].

Those contributions have led to the creation of several scholarly impact metrics. We will review some of them that are partly regrouped in [42]:

- C_{avg} stands for the average number of citations that an author's articles received
- *h index* of an author is an index defined as the number h of articles of the given author that have received at least h citations [37]. In other words, if an author has an h-index of 8, it means that 8 articles of the author have been cited at least 8 times. It has been the most commonly used metric in the last few years.
- *m index* is the median number of citations received by papers ranking smaller than or equal to h [7].
- *g index* is the highest number g of articles that have received at least g^2 citations [26]. It has been proposed in order to highlight articles with much more citations than what h index would suggest.

- *i-10 index* has been proposed by google and is defined by the number of articles with at least 10 citations [11].

These are only some of the metrics that have been proposed, yet the *h index* has been the most important metric as it has led to several other metrics that start from this one.

At a more general level, the impact of the scholar can be indirectly assessed through the use of a scientific ranking. Indeed, there are international and national rankings that establish a classification at the level of the conferences or journals that publish the scholars. Such ranking uses some – and usually multiple – impact metrics to build the order among the conferences and journals. As an outcome, they propose an order, from the most important to the least important, for the best conferences/journals showing how strong their impact is. Depending on which metrics have been used to compute the final classification, one ranking will confer more importance to some aspects, for instance the citation count, than others. The use of rankings can bring some recursivity since we will be more likely to use a ranking if its own impact is high. Some of the most famous rankings are the Scientific Journal Rankings (SJR) [64] and the CORE ranking [59].

As mentioned earlier, in the current work, we will try to find what are the key parameters that have a positive effect on the scholarly impact. In other words, which are the key factors that are going to lead to an increase in the scholar impact. We must now decide which metric/measure/notion to use to quantify the impact in the scope of this paper. For the sake of simplicity, we will use the number of citation to quantify the impact of the scholar. The main reason is that the number of citation is very often the core metric in any impact metric. However, the idea is to develop an approach that could be reusable in the future with another impact metric. For example, we want to build an approach that could be rerun later with, for instance, the CORE Ranking, instead of the number of citation, if someone wants to focus on this particular metric next.

2.2 Machine Learning

The goal of developing machines that would be able to imitate human's reasoning and making smart decisions was already present in the 20^{th} century [32]. The purpose of the works that have been made in this field was to bring machines into learning from past experiences in order to perform cognitive functions and solve complex problems [32]. In this section, we will be defining machine learning as precisely as possible. After that, we will be showing different models that exist and that will allow us to understand what kind of model is usable in our case.

2.2.1 Definition

Following what IBM says, Machine Learning is a part of Artificial Intelligence (AI) and computer science, that tries to replicate the way that humans learn through the use of data and algorithms, by trying to constantly improve its accuracy [23]. AI, according to [49], *"is the science and engineering of making intelligent machines, especially intelligent computer programs"*, while an algorithm is a sequence of instructions that have to be completed to transform the input into output [28].

So, Machine Learning is made to answer the way computers can *"learn"* determined tasks such as recognizing characters [29], classifying objects according to their characteristics,... To formulate it in another way, it deals with the question of how to create computer programs that are going to perform better and better through experience when completing a task [52]. It consists of programming computers using example data or past experience in order to optimize a performance criterion [28]. Machine learning can be predictive, in order to make predictions, or it can be descriptive to better understand and know the data, or both [28]. In such a way, all the data that have been collected can be split into two

subsets; the *training* dataset and the *test* dataset [67]. The Machine Learning model will first train itself with the training dataset. The latter refers to past data that have already been observed. So, the dataset is used to train the model, i.e. the model browses this data and learns from it, from the past experiences [80, 66]. Second, the Machine Learning model is tested with the test dataset to see how well it is performing. It is formally defined as *"a set of examples used only to assess the performance of a fully-specified classifier"* [9]. It will allow to evaluate the competing models, when a model is completely trained [66].

In addition to that, Machine Learning has two main branches: supervised learning and non-supervised, or unsupervised, learning [29]. These are the points that are going to be discussed in the following subsections.

2.2.2 Unsupervised Learning Model

Unsupervised Learning is a Machine Learning technique that can be seen as a *"cluster"* technique [65]. A cluster is defined in the Cambridge Dictionary [21] as *"a group of similar things that are close together"*. In this way of working, entities or observations that share similar behavior will be somehow grouped. In unsupervised learning technique, there is only input data and the goal is to find regularities in that input, there is no label or category to be predicted [29, 65]. Thus, unsupervised learning is a very effective approach when the goal is to uncover structures in populations of objects, when there is only little knowledge about the relationship of these objects [8]. In other words, it tries to learn to distinguish features and the associations that are made in the distribution of the data [32].

Hence, Unsupervised Learning is used when training datasets are not available [92]. The fact that, as IBM describes it, unsupervised learning is capable of analysing and clustering unlabeled datasets, and that it is mainly used to discover differences and similarities in information [25], makes it less interesting as the purpose of our work is different.

Indeed, unsupervised learning allows to, for example, discover groups of similar entities within the data, and is called clustering. [40] In other words, as google developers define it, clustering is the grouping of unlabeled examples [17]. Unsupervised learning, also allows to *"determine the distribution of the data within the input space"*, which is known as density estimation [40]. It also permits to project high-dimensional data on a computer screen [40], but none of it is useful in the case of the scholarly data we want to use. We are not interested in grouping similar scholar, but instead we want to predict a label, the impact, for a scholar that has a given set of features.

2.2.3 Supervised Learning Model

Supervised Learning is defined by Russel and Norvig in [62] as:

"The type of feedback available for learning is usually the most important factor in determining the nature of the learning problem that the agent faces. [...] The problem of supervised learning involves learning a function from examples of its inputs and outputs."

Supervised Learning is a Machine Learning technique that is also known as classification or identification technique, and consists of a labeled dataset that needs to identify unknown classes [65]. It is called *"supervised"* because the learning process is guided thanks to the presence of the outcome variable [81].

Supervised learning uses a provided input database which is then separated into training and testing datasets [67]. The supervised model aims to predict a label for each instance. To do so, it first learns the training data set, to understand how the label is attributed. Once the model is trained, it can be tested on the test dataset. An instance from the test dataset is given to the model without giving the label that is known. The model is asked to predict the label. The predicted label is compared to the real label, the one that should have been found. If both labels are not equal, this increments the error rate

of the model. Once the model reaches a determined performance level, it can be used to predict the label of a completely new observation, for which the label is unknown. While performing the prediction, the model needs to be given the right meta-parameters or hyperparameters. Hyperparameters are, as defined in [58], *"the variables which determine the network structure(Eg: Number of Hidden Units) and the variables which determine how the network is trained(Eg: Learning Rate)"*. Hence, they can be responsible for better performing models, but can also cause the prediction slow down and thus, they need to be chosen carefully.

Supervised Learning models are the exact types of models that fit the scholarly data. There are a lot of available techniques but we will explain those that seem to be the most interesting for our case.

K-Nearest Neighbors (kNN)

The K-Nearest Neighbor algorithm (kNN) is a non-parametric classification method [36], which is defined by [54] as *"a simple algorithm that stores all available cases and classifies new classes based on a similarity measure (e.g. distance functions)."* The principle is that a case is classified according to the majority of its neighbors and that the case is assigned to the most common class of its K nearest neighbors, measured by a distance function [54].

Among the advantages of this model, we can find in [83] that the algorithm is very simple and intuitive, and, as it is a memory based approach, kNN can easily adapt to new training data (lazy learning). However, the computational complexity of the algorithm increases very fast as nearly all the computation takes place at classification time [36][83]. Also, if the data used to train the model mainly represents 1 label, then this label will be more likely to be predicted, and thus there is a poor performance on imbalanced data [83]. Finally, the hardship is also hidden behind the selection of the optimal value of k, which can cause either underfitting or overfitting [36][83].

Overfitting is when the generalization of the model is unreliable in a sense that the model learns *"too much"* from the training dataset, and therefore there will be a lack of generalization [3]. Underfitting is the case where there is not enough learning made by the model and so, it makes it difficult to capture trends or to find patterns in the data if the model only knows a little about what is happening in the data [3].

Decision Tree (DT)

Decision Tree algorithm is a tree in which *"internal nodes can be taken as tests (on input data patterns) and whose leaf nodes can be taken as categories(of these patterns)."*[5]. The training process can be represented by a flow chart with internal nodes that test attributes and the resulting branches being outcomes of the performed tests [16]. The goal is to make conclusions about a sample through the observations of the created predictive model [16]. DT are split into two main categories: classification trees, that are used to predict the class to which a data sample belongs, and regression trees, which is used when the outcome is a real number instead of a classifier [16].

DT are really easy to interpret, and the data is easy to prepare [56]. It handles missing data and allows a lot of optimizing options while being able to handle both numerical and categorical data [87], but it is again hard to find the adequate depth of the tree to avoid underfitting and overfitting [87].

Random Forest (RF)

Random Forests are directly related to decision trees as they consist of a large number of decision trees, as [89] says, that work as an ensemble. So, each decision tree will give a prediction, and the class that occurs the most becomes the model's prediction [89]. According to [41], *"A large number of relatively uncorrelated models (trees) operating as a committee*

will outperform any of the individual constituent models.", and that is what makes random forests interesting. Also, still in [41], random forests will most of the time perform better than single decision tree classifiers. The advantage of this Machine Learning model is that it reduces overfitting in decision trees while helping to improve accuracy [89]. Indeed, the bigger the number of decision trees, the less the chance to pick a wrong class as the class that occurs the most is chosen. However, as a direct consequence, the inconvenience of such a solution is related to the amount of computational power it will require and the time that the training will take to be performed [89].

Support Vector Machine (SVM)

The objective of the Support Vector Machine is recalled in [34] as being *"to find a hyperplane in a N-dimensional space (N - the number of features) that distinctly classifies the data points"*. The goal is to find the best plane that makes two or more classes be the furthest from each other [34]. However, this model suits particularly binary classification, i.e. when there are two classes [8].

Among the advantages of this model, there is the fact that it aims to avoid overfitting while fitting the training dataset [8]. It has the advantages of *"increasing class separation and reducing expected prediction error [...] and it is suitable for analysis of high-dimensionality datasets with small sample size"* [86]. But the main disadvantage is that SVM is mainly suitable for small datasets [57].

Artificial Neural Networks (ANNs)

Artificial Neural Network is defined in [38] as *"the piece of a computing system designed to simulate the way the human brain analyzes and processes information."* ANNs use learning algorithms that are able, after receiving new input, to either learn or make adjustments [79].

ANNs have the ability to learn complex and non-linear relationships, while being able to generalize, and it doesn't impose restrictions on the input variables [48]. However, Neural Networks are a "black box" and are prone to overfitting while being difficult to use [82].

Regression

The regression is a method that is used to model a target based on independent predictors [33]. There are several types of regression but we will be explaining the simplest one, to give a more detailed view of this machine learning technique.

The linear regression, or multiple linear regression in this case, is a basic and commonly used type of predictive analysis [70]. It is one of the easiest and most popular Machine Learning algorithms that is used for predictive analysis [39] and is calculated by [45]:

$$\hat{y} = b_0 + b_1 X_1 + b_2 X_2 + ... + b_p X_p$$

with $\hat{y}$ being the prediction or estimation of the dependant variable, X_1,...,X_p being the independent variables that will be used to predict $\hat{y}$, b_0 being the value of $\hat{y}$ when all the independent variables are equal to 0 and b_1,...,b_p being the estimated regression coefficients, i.e. it gives the sensibility of the dependant variable in relation to the independent variable to which a given coefficient is attached [45].

In order to know if the prediction is successful and has worked in a correct way, linear regression has some indicators such as the R-Squared, which is a statistical measure that shows how close the data is to the fitted regression line [22], i.e. it explains how much the model's inputs can explain the observed variation from 0 to 100% [30].

This machine learning algorithm is really simple to implement and to interpret, but it over-simplifies real-world problems by assuming a linear relationship among the variables [61], and should thus be used carefully.

Chapter 3

Related Works

The research contributions of the current work got inspired from [47]. The authors of [47] first explain all the concepts included in the scholarly data and how they are structured. Then, they propose some visualizations of such data and the associated visualisation techniques. They conclude the work with some future researches in the field of scholarly data. One of these proposed to work on the co-author and co-citation relationship in order to assess which one has the biggest influence on the scholar impact. It directly led us on the track of the current topic, to assess the influence of all the features on the scholar impact, thus without restricting ourselves to only the co-author and co-citation relationships.

As mentioned before, a lot of work has been done in the field of scholarly data but we found very few works [53, 63, 84] that tackle concerns similar to ours.

[63] tries to predict the future number of time a scholar will be cited in another one. They created *FutureRank* that is a predicted rank based on the predicted citations of a scholar, the influence of its authors, and the precise moment of the scholar publication.

In a similar direction, [84] focuses on predicting the future impact of a scholar and its authors, especially the youngest authors with impact that is yet to be defined. The goal is to find, in advance, interesting new research areas.

In [53], an evaluation model is developed to incorporate more than only the number of citation as determinant of the scholar impact. It allows to better assess one's scholar impact to then predict its future impact. The ultimate goal is to predict the future *"Academic Rising Stars"* [53].

Even if some works focus on the scholar impact and some of its key drivers, the current work is different on several aspects. First, the current work focuses on understanding the influence of the key drivers in order to provide future users with recommendations enhancing their understanding.Also, we use here more recent data, trying to cover a lot of features, and we do not restrict it to specific fields.

Chapter 4

Methodology

In the following section, we will be discussing the methodology that has been applied in order to try to measure the features that influence the scholar's impact. First, we will be explaining the data that has been used and what were the challenges encountered when taking a closer look at it. Then, we will be explaining how the data has been prepared and cleaned, and what are the choices that have been made regarding the data. Finally, we will be showing the methods that were used, and more especially the Machine Learning models that have been trained, to achieve it. All data manipulations, the data preparation and the machine learning implementation have been done in Python. The code is available through this link: `https://github.com/wirex7/Master-Thesis.git`.

4.1 Data Sources

To address our initial question regarding the scholar impact, we will need scholarly data that gives us enough information on the articles that have been published. In addition to that, we will need to have datasets that have to be large enough in order to be able to find patterns. So, for this, we have used two datasets; the Citation Network Dataset and the SCImago Journal Rank Dataset.

(i) Citation Network Dataset [75, 73, 76, 74, 72, 68]

This dataset [4], that will be called CND for the rest of this work, is mainly extracted from DBLP, ACM, MAG (Microsoft Academic Graph), but also other sources, and contains data for a large number of papers (more than 4M). It is stocked under the 7-Zip format which can be opened as a JSON file [15, 51]. Each instance, i.e. each row, of this dataset concerns one unique paper, for which a lot of columns are available that describe some feature of the concerned paper such as its authors, the number of citations, etc., all of this for a total of 16 columns.

The different columns are:

- An **id**, that identifies the paper within the dataset;
- The **title** of the given paper;
- A column **authors** that contains all pieces of information about the paper's authors. For each author, **authors.name** is the name of the author, **authors.org** is the author's affiliation and **authors.id** gives a unique id for the author;
- The **year** of publication of the article;

- **n_citation**, which is the number of citation of the paper;
- The **doc_type** which gives the type of publication of the paper (i.e. book, journal, conference,...);
- The **publisher** of the doc, which gives the name of the organization that published the paper;
- A column **volume** and a column **issue**, that are defined in [6] as "volume typically refers to the number of years the publication has been circulated, and issue refers to how many times that periodical has been published during that year";
- The **doi** of the article, that stands for Digital Object Identifier (DOI) and that is defined by "a unique and never-changing string assigned to online (journal) articles, books, and other works" [12];
- The **references** of the article, that gives all the sources that are cited by the article;
- The **page_start** and **page_end** that are the beginning page and the end page of the article in the doc_type it is published;
- The **indexed_abstract** that is made of the indexed_abstract.IndexLength that gives the length of the abstract in terms of words, and the **indexed_abstract.InvertedIndex** that gives the position and the occurence of each word of the abstract;
- The **fos** which is the field(s) of study, extracted from MAG. For each field of study, **fos.name** is the field of study itself and **fos.w** is the weight of the given field of study reflected in the paper;
- A **venue** which is made of the **venue.id** and the **venue.raw** that respectively give the id and the name of the conference or journal where the paper has been published.

This dataset contains a lot of information, referred to as features in the remainder of the paper, about the scholars. This is our main data source that will also be complemented with other pieces of information throughout the current work. Therefore, this dataset will be, in the next sections, subject to data manipulation and data preprocessing in order to get the data prepared to be used with a well-known Python Machine Learning Library, namely Scikit-Learn [46].

(ii) SCImago Journal Rank Dataset [scimagojr.com]

This dataset, that will be called SJR for the rest of this work, is extracted from the SCImago Journal & Country Rank [64], and contains the journals and country scientific indicators, with more than 34k instances. It is stocked under the Excel format which makes it easy to use and to transform. Each instance of this dataset concerns one unique conference/journal for which a lot of columns are available that describe its statistics, and that calculate its ranking. A detailed description of the calculation of the ranking is given by [35].

This dataset is made of 20 columns that are:

- The **Rank** of the Journal/Conference/Book (J/C/B) **Title**, according to the **SJR** calculation;
- The **Title** which is the name of the J/C/B;
- The **SJR** which is the SCImago Journal Rank indicator and is the "measure of journal's impact, influence or prestige. It expresses the average number of weighted citations received in the selected year by the documents published in the journal in the three previous years" [64];

- The **H index** that is given by the "Journal's number of articles (h) that have received at least h citations over the whole period;
- A **Sourceid** that refers to a unique J/C/B;
- The **Type** of the publication that can be either a Journal, a Conference or a Book;
- The **ISSN** which is an identifier for the J/C/B (also defines a paper);
- The **SJR Best Quartile** that is self explanatory;
- **Total Docs. (2020)** and **Total Docs. (3years)** that give all the published articles of the year and the 3 previous years;
- **Total Refs.** which is the number of references included in the articles that are published by the J/C/B;
- **Total Cites (3years)** which gives the citations in 2020 that are received by journal's documents published in the three previous years;
- **Citable Docs. (3years)** that depicts the journal's citable documents (i.e. articles, reviews and conference papers) in the three years before 2020;
- **Cites / Doc. (2years)** which gives the average citation per document in a 2 years period and is commonly used to determine the impact of a J/C/B;
- **Ref. / Doc.** that gives the average amount of references per document;
- The **Country** and the **Region** concerned;
- The **Publisher** which is the same as for the previous dataset, and so determines the owners of the J/C/B;
- The **Coverage** that gives the period of publication of the J/C/B;
- The **Categories** that determine the domains that concern the J/C/B;

The SCImago Journal data source comes as a complement of our main data source. It provides a lot of impact metrics that are lacking from the CND data. In the next steps of the current work, we will need to somehow assess the impact of the scholars. The SJR provides us with metrics that can play the role of "impact measure". SJR gives such measures that are linked with a Journal/Conference/Book that can be easily linked with the scholars of the first database (refer to section 2.1.1 Database Creation). The SJR data sources also provide additional pieces of information, that will ultimately turn into paper's features, like for instance the Coverage, some geographical information, and so on.

4.1.1 Database Creation

Both datasets can be used in several manners when merging them together. More precisely, the second dataset provides the H index, which can be very useful to measure the scholar impact, as we mentioned previously. Also, it can allow us to measure the impact of a Journal/Conference/Book on a given article through the ranking of the J/C/B, and thus, find out the causal relationship between the paper's features and the Impact. Therefore, to achieve this, a join has been made between the two datasets on their common column called "**venue.raw**" in the first dataset, and "**Title**" in the second one. In addition to that, both datasets have some identical columns such as the "**Type**" and the "**Publisher**". A description of all columns resulting from the join is available at Annex 7.1.1.

4.2 Pre-Processing

Before starting to deal with the newly merged dataset and to run machine learning models on it, the data needs to be prepared in order to fit the requirements of these models. Hence, the data requires a preprocessing, which is defined by [55] as:

"that step in which the data gets transformed, or Encoded, to bring it to such a state that now the machine can easily parse it. In other words, the features of the data can now be easily interpreted by the algorithm."

So, the data is going to need to get cleaned, but it will also need some feature engineering which consists in creating new feature columns from the data that is at our disposal. This is what we are going to discuss in the upcoming points. In the section Data Cleaning, we present all the issues we faced regarding the data preparation. To solve some of them, we had to create new features that we describe in the section Feature Engineering.

4.2.1 Data Cleaning

Regarding the cleaning of the data, the first dataset contained some tricky problems that needed to be tackled.

Issue 1: Access to the Data and Performance

First of all, the number of instances was way too high, and the format was problematic as it required much more memory than a csv-file for example. Hence, the kernel of the program crashes each time it tries to assign the data to a variable. In order to face this problem, the solution was to split the data in a random way, by creating smaller json files that could be opened. However, it required to split the data into 40, and so, taking only one file was making the data insufficient. So, the solution was to open the json file, transform it so that a csv file containing its data can be obtained, and then repeat the process until enough instances were transformed into a csv file. After that, we needed to open the different files again on a program and then merge them as a csv file. At the end, we found out that parsing the same volume of data in csv was much faster and required way less memory than a json file.

Issue 2: Multivalued Features

Once the data could be utilized easily, some problems related to the structure of the columns of the data appeared. For the first dataset, some columns like the "**authors**" one contained a data structure encapsulated into another data structure. Instead of having a single value as recommended by the theories of relational databases, this column contained a list of dictionaries that needed to be treated before using it. Indeed, running models on such kinds of multivalued columns is very hard and inconvenient, as the machine learning models favour columns that contain one single value per instance. We faced a lot of columns with the multivalued problem, namely the columns "**authors**", "**indexed_abstract**", "**fos**", and "**venue**". To counter these problems, we implemented several solutions. For example, we removed some of the problematic features, we extracted the data in order to make single-value columns, we made dummies columns, we aggregated the multi-values into one, we decomposed the encapsulated data structure into an atomic one, etc. For the relevant columns, we will explain how we handled them in the section Feature Engineering.

Issue 3: Missing Values

After that, columns such as "**reference**" contained missing values and missing references, and some part of the data contained a column "**alias_ids**" that contained a lot of Non-

attributed values, and so we had to get rid of this column, as it was not even discussed in the user-guide of the source of the dataset. Finally, the "**author.org**" was not regular as values were changing significantly from one author to another, which made it difficult to extract interesting information. Other missing values were treated either by removing the corresponding row, or by imputing a placeholder value (like the mean, the median, etc), depending on the number of row implied and the feature relevance.

Issue 4: Merge of Data Sources

Also, the merge was problematic between the two datasets as the common column that was used sometimes contained values that were slightly differing (MAJ instead of min, missing term, ...) and so, the number of instances decreased heavily. To deal with it, we manually adapted the values of the second dataset to make most of them match with the first dataset.

Issue 5: Textual Features

Apart from that, one of the main problems was about all the columns that were not containing numerical values. Indeed, the majority of Machine Learning models, used in this work, were requiring numerical features. Of course, it is possible to run Machine Learning tools with textual features especially by using Text Mining tool, but this would have asked a lot of others technologies and tools like uploading Linguistic Taxonomies, libraries that handle words occurrences and context analysis, and Machine Learning models that particularly focus on the Text Mining branch of the Machine Learning discipline. While all of this is possible, and may be highly powerful, this is not the ambition of this project. This is why we had to find some strategy to solve the Textual Features and we found different solutions; removing the feature, transforming the textual feature into numerical features, creating dummy-encoded columns, etc. For the concerned columns, we will specify the chosen strategy in the section Feature Engineering.

Issue 6: Quantitative and Categorical Label

The last issue, before having a dataset ready to be used, was to prepare the "label" column. The label column is the feature that the Machine Learning Model will have to predict. In our case, this is a metric that is going to determine the impact of the scholar, given by the column "**n_citation**" which is the number of times the paper is cited. The higher the number of citations, the more impactful the article will be. In this particular case, the label is a quantity which is very suitable for quantitative models (like the SVM, the Regression, etc). However, a lot of models, like the KNN, the Decision Tree, the Random Forest, etc, require the label to be a category (for instance category A, category B, category C, etc) and not a quantity. In order to be able to use such models, we had to somehow transform the label "**n_citation**" into a category. The solution was to create some bins, i.e. some categories or intervals, that contained a range of discrete values. But it raised a lot of questions: how many value must be taken within one bin? How many bins do we have to create? How to choose between a fixed interval length or a fixed number of occurrence in each bins? Fortunately, a lot of mathematical theories exist to properly handle this kind of problem and, therefore, to break this problem down, we relied on a formal tool proposed by SKlearn [46], the KBinsDiscretizer [20]. We further elaborate on the tool in the section Feature Engineering.

4.2.2 Features Engineering

Based on the columns that we initially had and the problems we faced, we created several new features.

Feature 1: Pages

First, we created the "**pages**" column that takes the "**page_start**" and "**page_end**" of the given article and calculates the number of total pages.

Feature 2: Authors

Second, we created the column "**nb_authors**" based on the initial "**authors**" column to calculate the number of authors that have worked on the paper.

Feature 3: Abstract

Then, we used the "**indexed_abstract**" column to extract the length of the abstract and the occurrence of the different words, to measure the impact each of these parameters has on the number of citations.

Feature 4: "Main_FOS" and "First_Common"

After that, we focused on the "**Main_FOS**" and "**First_Common**" columns. These features respectively correspond to the main Field of study, i.e. the field of study of the article with the higher weight, and the most common word of the abstract (excluding the *"stop"* words like the specifiers, the pronouns, etc). These features are, of course, textual features. However, as mentioned with *Issue 5* in section Data Cleaning, we wanted to avoid as much as possible such features. To do so, instead of using the "words" themselves, we decided to use the frequency of the word in the whole dataset. For instance, the word *"Cluster Analysis"* was the most frequent *"Main_FOS"* in the dataset and it has been observed 1347 times. We then replaced the value "Cluster Analysis" by the value "1347" for all concerned rows. We applied this manipulation for both columns and so, it allowed us to transform two textual features with two numerical ones by still integrating the abstract and field information into the dataset. Now, the new features express a notion of "popularity" regarding the main Field Of Study and the most frequent word of the abstract instead of the word itself. Such columns are still relevant since they could lead to recommendations like for instance "does the use of popular words in the abstract influence the scholar impact or not?" etc.

Feature 5: *"doc_type"*

The column "**doc_type**" can take two values: "**Conference**" if the scientific production is published within a conference, or *"Journal"* if the scientific production comes from a journal. As explained in the Data Cleaning section, we want numerical values for the remainder of the work. Hence, we used a tool called *OneHotEncoder* from the SKlearn library [46] as it allows to transform categorical or textual value into numeric (or boolean) value. In our specific case, all "conference" values have been transformed into "1", and all "journal" values have been transformed into "0". This allows to keep the publication type information but with now numerical values.

Feature 6: *Categorical Label*

As explained in section Data Cleaning, we had, at some point, to create a new feature which is a categorical label expressing the number of citations. We called it "**KB_cat**" and the resulting column is shown in the Annex 7.2. To do so, we used the *"KBinsDiscretizer"* [20] of the SKlearn library [46]. This tool constructs some bins, some categories, with a quantitative input. We had to specify some parameters including in particular:

- *"n_bins"* : the number of bins that have to be created. We tested this parameter iteratively. First, we set a low value to see how the dataset reacted. This led us to the number of 20 bins.

- *"strategy"* : the strategy to use to create the bins. It can be; *"uniform"*, *"quantile"* and *"kmeans"*. The two first values are relevant when constructing bins on several features, which is not our case. The last value creates the bins of one feature following a K-Means Clustering mood which fitted what we wanted to do.

Feature 7: *Countries*

Here again, the column "**Country**" is a textual feature. To deal with it, we created what is called some "dummy-encoded columns". In the initial dataset, each row, i.e. each paper, had one value, for instance "Belgium", in the column "Country". In order to get numerical value, we created one column for each unique value of the *"Country"* column. If we take back the example, the value "Belgium", which was initially a value in a column, is now an entire column itself called "Belgium". Now, each paper that is published in Belgium, will get the value "1" in the column Belgium. If the paper is published somewhere else, it will get the value "0" in the column Belgium and the value "1" in another column that represents a country. The result of this manipulation is shown in Annex 7.1.4. This solution was feasible since we had only 38 different countries meaning that we created only 38 new Boolean columns. We tested the models of the section Machine Learning Model, first, without the dummy-encoded columns, and then with the dummies. Since it did not seem to affect the performance, we decided to keep this solution alive.

4.2.3 Final Datasets

Thanks to all those modifications, we have created two different datasets. Both have the same amount of columns (49) among which 48 are the same, but one is different.

Indeed, we have used the "*n_citation*" label from the initial dataset for the first dataset that we have created because it is going to be used for quantitative models as it is a quantity, so we kept the same values. But, for the second dataset we created the "*KB_cat*" label, which splits the initial values of the "*n_citation*" column into categories. So, for example, the first category contains the papers that are referenced from 0 to 20 times, the second contains the papers that are referenced from 20 to 58 times,... as we can see in the Annex 7.2.

The final state of the database is shown in the Annex 7.1: Annex 7.1.2 presents the columns of the dataset with the categorical label, Annex 7.1.3 presents the columns of the dataset with the quantitative label and Annex 7.1.4 presents the country dummies columns that are present in both datasets.

4.3 Machine Learning Models Training and Testing

After the cleaning of the data and the creation of new features, the data was ready to be fed to the Machine Learning Models. Based on the section Machine Learning of the current work and some tools like the guide *"Choosing the right estimator"* of SKlearn [18], we decided to focus on the following models: a KNN, a Decision Tree, a Random Forest, a SVM, a Regression, and an ANN. All the following manipulations were done using the Machine Learning Python Library SKlearn [46] which proposes a lot of different machine learning models and tools for such situations. It was chosen for its variety of tools and its strong community that provides a lot of documentation and support in case of issues. The first thing to do, like any machine learning training, was to split all the data we had into a training dataset, and a test dataset. We did that using a SKlearn tool [46]. We split our dataset to get:

- 80% of training data
- 20% of test data

We then worked in an iterative way. We created several versions of the dataset by varying the dataset size. We first took a dataset, with the same columns, but with a very low number of rows, namely 2000 rows. We wanted to test a first model on a lower scale to see how the model would react to the data. Each time the test was successful, we increased the size to get a model on a greater scale. It allowed to keep control and to solve rapidly errors that would raise. For each model and dataset size, we first run a "simple" model, i.e. without specifying any meta-parameters. Each time the "simple" model worked properly, we focused more into details to these meta-parameters to find the best possible value. For each configuration (i.e. a Machine Learning model, a dataset size and a configuration of meta-parameters), we collected and reported some metrics (like the used memory, the errors encountered, the time it took, the accuracy of the model, the problem regarding the data design, etc). All of this allowed us to refine all along our approach, first regarding the fine-tuning of the models, second regarding the preparation of the data (what we have reported in previous sections), and third regarding the limit of the technology we had at our disposal. Indeed, during this cycle of testing, we faced technological issues. When it was possible, we implemented strategies to counter the technological issues otherwise we did within the technological limits.

Finally, we reached the full dataset size and ran the final models. The strategy was simple: as already mentioned, we first trained a simple model on the training dataset. Then we tested the model using the test dataset. Concretely, we gave the test dataset to the model without giving him the label. Then, the model predicted the label for all papers in the testing dataset. We then compared the "real" label with the "predicted" one to assess the accuracy of the model. It gave us a metric, the accuracy, to assess how well the model performs. Then, we ran Grid-Search [19] to test multiple values of the meta-parameters of the model. Grid-Search is a SKlearn tool, that will run, for a given machine learning model, various configuration of meta-parameters values within the range we gave. This took time but allowed us to find values for relevant meta-parameters that would increase the accuracy. We also reviewed other parameters of our configuration to see if the accuracy could have been still increased. For instance, for models like Regression or KNN, we normalized the data. Once we were done with a model, we reported in the next section the model specification and results and we went to the next model with the same strategy.

Chapter 5

Results: Machine Learning Models Features and Performance

The very first step, in order to find out the key drivers of scholar impact, is to construct machine learning models that perform quite well when predicting a scholar impact based on initial scholar features. The idea is to get a model that can predict a scholar impact to then be able to extract the knowledge of the model that allows it to predict. When exploring this kind of knowledge, we would be able to understand which patterns in the features lead to defining the impact. This is what we have done and, as the tests progressed, we decided to diversify our implementation strategy. Indeed, instead of restricting ourselves to one model, we tested various models to assess which ones would fit the best the scholar situation. Each model having its own advantages, diversifying the models would allow us to see the benefits and the downsides of each of them. In what follows, we assess the performance of each model. The results of all the models are summarized in table 5.1. In the following paragraphs, we further elaborate on these results.

We are quite confident in these results because, for a first attempt, we did achieve to reach performance levels around 70% and a R^2 of 0.05 for the regression, which gave us hints of where does each model perform the best. Indeed, the regression, as told previously, allows to predict a quantity given some parameters. However here, the huge variation of the number of citations (going to more than 20 000), and the fact that the parameters chosen are clearly not the only ones responsible for the number of total citations of an article, makes it less relevant to use (see Annex 7.3.4). Also, the purpose was to find an "impact" which is better determined by categories instead of a precise number.

In general, we faced a trade-off between the "easy implementation of the model" (in other words, the easiness to find the optimal meta-parameters, to reach a strong accuracy level, and so on) and the "knowledge extraction". For example, a KNN model is quite simple to implement but lacks some knowledge extraction and can be pretty slow. Hence, it is not possible to access the pattern that is behind the model predictions, or to understand which features have worked towards reaching the given accuracy. Another example, is the use of ANN. While they can be very powerful, it takes too long to explore the meta-parameters to find the right values[1].

For some preliminary works, we are confident to see that good accuracy level can be reached. We are aware that a good accuracy level is not sufficient. But we believe that, at a greater scale, we can have both a performing model that can provide us with knowledge

[1]Note that the column "time" of table 5.1 refers to the time it takes to run the final model, not the time it takes to find the best configuration of the model.

regarding its underlying patterns.

In addition to the models of the table 5.1, we also ran a SVM model. Since the model starts struggling when the data is hard to split with lines and since it becomes rapidly slower as the number of dimensions increases, we did not integrate it into the results. Indeed, simply running the model without modifying any parameter took more than 6 hours and crashed several times. This poor performance is not a surprise. SVM models usually perform well with situation where there is few categories to predict and few features, which is absolutely not the case with our data.

Model Type	Meta-Parameters	Accuracy	Time	Label	Data Size	Advantages	Downsides
KNN	• n_neighbours = 119 • weights = 'distance'	68.5%	14min	KB_cat	351588 rows × 49 columns	• Easy to implement • Good accuracy	• Poor interpretation • Impactful features remain unknown • Slow calculation
Decision Tree	• max_depth = 12 • max_leaf_nodes = 149	69%	5sec	KB_cat	351588 rows × 49 columns	• Very fast calculation • Good accuracy • Allows to extract features with their impact • Easy to interpret	• Hard to find good meta-parameters
Random Forest	• max_depth = 35 • max_leaf_nodes = 1900 • n_estimators = 300	69%	5min	KB_cat	351588 rows × 49 columns	• Good accuracy • Allows to extract features with their impact • Easy to interpret	• Hard to find good meta-parameters • Slower than the the decision tree
Regression	• shuffle = False	R^2 = 0.05	1sec	n_citation	351588 rows × 49 columns	• Very fast • Gives the possibility to see the coefficients	• Inappropriate in our case • Tries to find an exact value
ANN	• max_iter=300	69%	13min	KB_cat	351588 rows × 49 columns	• Good accuracy	• Becomes very slow when trying meta-parameters • Hard to draw any conclusion

Table 5.1: Machine Learning Model Summary

5.1 Recommendations to Future Users

These results allowed us to bring some recommendations to any future user that wants to go further in a certain aspect of this work. For example, if we take only one Machine Learning model such as the Decision Tree, it allows an extraction of the rules (i.e. the knowledge of the model) that were used to predict the label. Also, it allows to have an idea of the coefficients of each feature that was used to make the prediction, as we can see in the annex 7.3.2. Given this, we can then see the way the model made prediction regarding a new instance, as it is shown in the annex 7.3.1.

Given the results of the Decision Tree model, we can see that the H-index is the highest value, meaning that its the parameter with the highest impact on the label, as explained in [69]. In other words, this parameter is the one that is in charge of the highest gain in information, leading to the classification of a given instance in certain categories. We can also see in 7.3.2 that a higher number of pages and a higher number of references can lead to a better cited article.

The same reasoning, the knowledge exploration and the recommendation extraction, can be done with various other models.

Chapter 6

Discussions

6.1 Limitations

There are some limitations to this work, but most of them are not impossible to resolve.

First of all, the technology and the tools used for this project bring some restrictions that are very bothering. In fact, dealing with a huge amount of data, and running machine learning techniques on it requires a large RAM (Random Access Memory). The bigger the dataset, the harder to use it all. In our case, the initial dataset had more than 3 Million instances, but only 350 000 instances could be extracted as it was the limit of the memory of the computer. Yet, it is a very reasonable amount, but with a larger memory many more operations and calculations could be made easily.

Second, we did not focus particularly on the years of the publications. It would have been very interesting to know how the dataset varies across the years, are the papers evenly distributed among years or are they big differences,... In addition to that, the results would be very interesting if we managed the years as most of the results provided by the *SJR* dataset focus on the 3 years preceding the year selected while downloading the dataset. So, with a much larger amount of instances from the *AMiner* dataset, we could use SJR dataset matched with the exact years of the *AMiner* dataset to get much more information about the journals, the citations throughout the years,... This again will require much better performing RAM for a computer.

Also, we tackled this problem dealing with quantities as it fits much better the requirements of Machine Learning techniques, but it seems evident that only the characteristics that we treated could not be exactly responsible of the final number of citations a paper can get. There are many other parameters related to the content, the words used,... that would have helped to determine the impact of a scholar. We could work from a text mining perspective, and even try to transform columns into dummies, as we did with the Country one, but it would again require much more RAM to be done.

Apart from that, we could also use other datasets, as there are many other that may provide interesting information and help to tackle the problem from different angles.

6.2 Future Research

This work provides, according to us, a very interesting basis for future works.

By overtaking the limitations that were previously cited, it would allow to produce much more consistent results with a much higher level of accuracy. Indeed, the preliminary results are very promising and show that it would be valuable to bring this topic to further levels and to work on it at a greater scale. For example, transforming each of the textual columns that were in the initial datasets in a form that is usable for the Machine Learning techniques, would be a complete work on its own.

Also, work could be made on the relation between the papers and the authors, as the references and the authors are identified by id's. This would allow a better understanding of the impact that a given author or reference may have on an article.

The work emphasizes that having a kind of recommendation tool for researchers and academics is still relevant. This exploratory work shows a real opportunity regarding the feasibility of such tool.

6.3 Conclusion

This work was focused on the Scholarly Data. First, all the necessary background has been detailed regarding the Scholarly Data and Machine Learning fields ensuring great understanding of the remainder of the work. Then, we reviewed the work that has been done in similar fields.

After that, we started exposing our methodology and result. The objective was to explore Scholarly Data in order to clarify which feature does influence the scholar impact. In order to find out, we ran several machine learning models: a KNN, a Decision Tree, a Random Forest, a Regression and an ANN. For each of them, we refined the configuration (the meta-parameters, the type of the label, etc) in order to get strong accuracy, and to reduce potential underfitting/overfitting. In general, these models perform well, being around 70% of accuracy. With one of these models, the Decision Tree, we showed how it is possible to extract the knowledge of the model in order to translate it into concrete recommendations. These would support any new scientific writer on which features matter the most.

At the end, the purpose of this work is to clearly show that these results are encouraging to go further in this direction as the dataset can be used with several machine learning techniques. Any model can be chosen regarding the convenience and the expectations of any unique user, the goal is not to limit them, but instead to give an overview of the possibilities that the scholar data gives.

Finally, the limitations and the future researches are presented and they both insist on the fact that this exploratory work may be continued.

Chapter 7

Appendices

7.1 Datasets Snapshots

7.1.1 Original Data from both Datasets

Col name	Description	Data type	Data origin	exemple
id	identifier of the paper	string	Citation Network Dataset (CND)	“2022950330”
authors	a detailed description of the author	list of dict	CND	[{‘name’:’Corinna Cortes’, ‘org’: ‘AT&T Bell’,’id’: 2134830209}]
title	the title of the given article	string	CND	‘Support-Vector Networks’
year	year of the publication	integer	CND	1995
n_citation	number of times the article has been cited	integer	CND	22276
doc_type	type of publication	string	CND	‘Journal’
publisher	publisher of the article	string	CND	Kluwer Academic Publishers
volume	number of years the article has been circulated	integer	CND	20
issue	number of times the periodical has been published that year	integer	CND	3
doi	identifier assigned to the outside of the dataset	string	CND	10.1023/A:1022627411411
references	the id’s of the references of the article	list	CND	[2087347434, 2154579312, 2168228682]
indexed_abstract	the abstract length and content	dictionary	CND	{‘IndexLength’:122, ‘InvertedIndex’: {‘data’:1}}
fos	the fields of study of the paper and their weights	list of dictionary	CND	[{‘name’: ‘Online machine learning’, ‘w’:0.617}]
venue	the paper’s	dictionary	CND	{'raw': 'Machine

	publication place (i.e. the journal/conference/book)			Learning', 'id': 62148650, 'type': 'J'}
Title	extracted venue.raw of the conference	string	Created from CND	‘Machine Learning’
rank	the rank of the Journal/Conference/Book (J/C/B)	int	SCImago Journal Rank (SJR)	7505
Sourceid	defines a unique J/C/B in the database	string	SJR	‘24775’
Type	type of publication	string	SJR	‘journal’
Issn	identifier or a J/C/B outside of the data	string	SJR	‘15730565, 08856125’
SJR	the measure of journal’s impact	float	SJR	0.667
SJR Best Quartile	best quartile of the SJR	string	SJR	‘Q1’
H index	journal’s number of articles h that have received at least h citations	integer	SJR	152
Total Docs.(2020)	number of published articles of the year	integer	SJR	87
Total Docs. (3years)	number of published articles of the last three years	integer	SJR	233
Total Refs.	total of references of the articles published in the J/C/B	integer	SJR	4124
Total Cites (3years)	number of citations that are received in the current year by the articles of the three previous years	integer	SJR	1104
Citable Docs. (3years)	number of journal’s citable docs in the three previous years	integer	SJR	218

Cites / Doc. (2years)	Average number of citations per document	float	SJR	4.59
Réf. / Doc.	average number of references per document	float	SJR	47.4
Country	Country of the J/C/B	string	SJR	‘Netherlands’
Region	Region of the J/C/B	string	SJR	‘Western Europe’
Publisher	publisher of the document	string	SJR	‘Springer Netherlands’
Coverage	years of publication of the J/C/B	string	SJR	‘1986-2020’
Categories	categories of the J/C/B	string	SJR	Software (Q1); Artificial Intelligence (Q2)
pages	number of pages of the article	integer	created from CND	25
nb_authors	number of authors of the article	integer	created from CND	2
author1	name of the first cited author of the article	string	created from CND	‘Corinna Cortes’
index_length	the length of the abstract in terms of words	integer	created from CND	122
most_common	the four most common words of the abstract	list	created from CND	['support-vector', 'learning', 'network', 'surface']
first_common	the most common word	string	created from CND	‘support-vector’
second_common	the second most common word	string	created from CND	‘learning’
third_common	the third most common word	string	created from CND	‘network’
fourth_common	the fourth most common word	string	created from CND	‘surface’

7.1.2 Dataset with Categorical Label

	year	doc_type	volume	issue	H index	Total Refs.	pages	nb_authors	index_length	fc_pop	nb_references	fos_pop	KB_cat
0	1995	0	20.0	3.0	152.0	4124.0	25	2	122.0	1	3.0	47	18.0
1	2009	0	25.0	16.0	390.0	29619.0	2	9	89.0	14	8.0	3	17.0
2	2003	0	3.0	3.0	230.0	13524.0	30	3	120.0	376	11.0	87	16.0
3	1993	0	14.0	11.0	188.0	14925.0	17	13	96.0	13	2.0	5	15.0
4	1998	0	16.0	8.0	236.0	9052.0	8	1	89.0	44	3.0	59	14.0

7.1.3 Dataset with Quantitative Label

	year	n_citation	doc_type	volume	issue	H index	Total Refs.	pages	nb_authors	index_length	fc_pop	nb_references	fos_pop
0	0.879518	0.000000	0.0	0.000000	0.000022	0.008157	0.000463	0.000465	0.000000	0.123616	0.005396	0.002427	0.020802
1	0.855422	0.000943	1.0	0.004555	0.000033	0.123165	0.006923	0.001161	0.000000	0.044280	0.020614	0.043689	0.039376
2	0.891566	0.000449	0.0	0.003145	0.000022	0.076672	0.000706	0.001084	0.017544	0.025215	0.048700	0.014563	0.057949
3	0.855422	0.001841	0.0	0.001302	0.000000	0.054649	0.001423	0.002555	0.007018	0.080566	0.024073	0.038835	0.028975
4	0.746988	0.003995	1.0	0.002928	0.000033	0.048940	0.000000	0.000619	0.000000	0.135916	0.000553	0.031553	0.457652

7.1.4 Countries Dummies

Australia	Austria	Brazil	Canada	Chile	China	Czech Republic	Denmark	Egypt	France	Germany	Hong Kong	Hungary	India	Indonesia	Ireland	Italy	Japan	Netherlands
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

7.2 Categorical Label

KB_cat	lower boundary	higher boundary
0	0	19.78
1	19.78	57.82
2	57.82	125.16
3	125.16	239.94
4	239.94	438.13
5	438.13	774.31
6	774.31	1289.60
7	1289.60	2039.57
8	2039.57	3128.93
9	3128.93	4764.75
10	4764.75	6770.80
11	6770.80	8705.08
12	8705.08	9996.25
13	9996.25	10989.25
14	10989.25	12363.25
15	12363.25	16254.00
16	16254.00	19777.00
17	19777.00	21219.00
18	21219.00	22276.00

Table 7.1: Categories of n_citations

7.3 Machine Learning Models Results

7.3.1 Decision Tree Performance

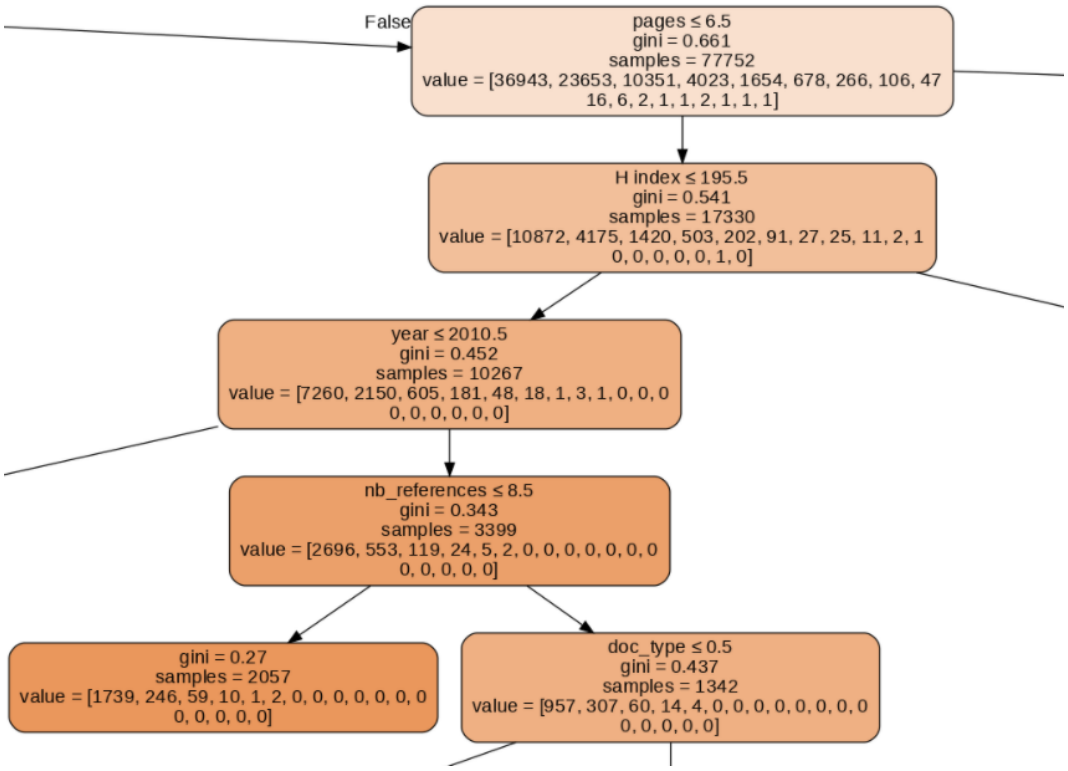
Actual	Predicted	
0.0	0.0	45890
1.0	0.0	11941
2.0	0.0	3461
1.0	1.0	2553
0.0	1.0	1984
2.0	1.0	1463

7.3.2 Decision Tree Coefficients

```
[(0.09998968550251497, 'year'),
 (0.0058995247973681955, 'doc_type'),
 (0.03319299604489986, 'volume'),
 (0.0025740763640076754, 'issue'),
 (0.4938029350188077, 'H index'),
 (0.05025954296371451, 'Total Refs.'),
 (0.20202705335450227, 'pages'),
 (0.0008137989757503929, 'nb_authors'),
 (0.0009191694935301629, 'index_length'),
 (0.0, 'fc_pop'),
 (0.0921974961993888, 'nb_references'),
 (0.0, 'fos_pop'),
 (0.0, 'Australia'),
 (0.0, 'Austria'),
 (0.0, 'Brazil'),
 (0.0, 'Canada'),
 (0.0, 'Chile'),
 (0.0, 'China'),
 (0.0, 'Czech Republic'),
 (0.0, 'Denmark'),
 (0.0, 'Egypt'),
 (0.0, 'France'),
 (0.0009949738252270418, 'Germany'),
 (0.0, 'Hong Kong'),
 (0.0, 'Hungary'),
 (0.0, 'India'),
 (0.0, 'Indonesia'),
 (0.0, 'Ireland'),
 (0.0, 'Italy'),
 (0.0, 'Japan'),
 (0.004013331570191456, 'Netherlands'),
 (0.0, 'Poland'),
 (0.0, 'Portugal'),
 (0.0, 'Russian Federation'),
 (0.0, 'Serbia'),
 (0.0023171344830635984, 'Singapore'),
 (0.0, 'Slovenia'),
 (0.0, 'South Africa'),
 (0.0, 'South Korea'),
 (0.0, 'Spain'),
 (0.0, 'Sweden'),
 (0.0, 'Switzerland'),
 (0.0, 'Taiwan'),
 (0.0, 'Turkey'),
 (0.0, 'Ukraine'),
 (0.0, 'United Arab Emirates'),
 (0.0, 'United Kingdom'),
 (0.010998281407033416, 'United States')]
```

7.3.3 Decision Tree Structure

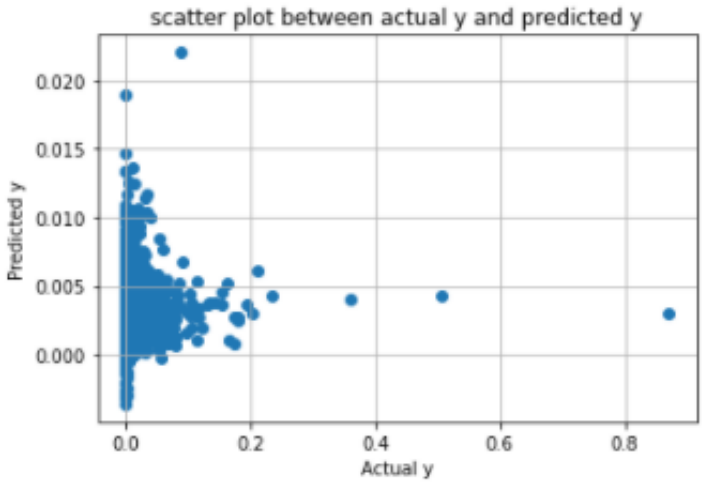
Part of the Full Structure



Link to the Full Structure

The full structure of the Decision Tree can be downloaded here: `https://drive.google.com/file/d/1AnhWzq8xxw7nS3WtxBtEuQEwdvyL4eW2/view?usp=sharing`

7.3.4 Regression Performance



Bibliography

[1] Herman Aguinis et al. "Scholarly impact: A pluralist conceptualization". In: *Academy of Management Learning and Education* 13.4 (2014), pp. 623–639. ISSN: 1537260X. DOI: 10.5465/amle.2014.0121.

[2] Herman Aguinis et al. "Scholarly impact revisited". In: *Academy of Management Perspectives* 26.2 (2012), pp. 105–132. ISSN: 15589080. DOI: 10.5465/amp.2011.0088.

[3] Anas Al-Masri. *What Are Overfitting and Underfitting in Machine Learning?* 2019. URL: https://towardsdatascience.com/what-are-overfitting-and-underfitting-in-machine-learning-a96b30864690.

[4] AMiner. *Citation Network Dataset*. 2021. URL: https://www.aminer.org/citation.

[5] Arundhati Navada et al. "Overview of Use of Decision Tree algorithms in Machine Learning". In: (2011).

[6] Theresa Bell. *What is a volume/issue number?* 2019. URL: https://writeanswers.royalroads.ca/faq/199175#:~:text=The%20difference%20between%20the%20numbers,2)..

[7] Lutz Bornmann, Rüdiger Mutz, and Hans Dieter Daniel. "Are there better indices for evaluation purposes than the h index? A comparison of nine different variants of the h index using data from biomedicine". In: *Journal of the American Society for Information Science and Technology* 59.5 (2008), pp. 830–837. ISSN: 15322882. DOI: 10.1002/asi.20806.

[8] Anthony Brabazon, Michael O'neill, and Seán Mcgarraghy. *Natural Computing Series Natural Computing Algorithms*. 2015. URL: www.springer.com/series/.

[9] Brian D. Ripley. *Pattern recognition and neural networks*. 1996.

[10] Prince George's Community College. *Bibliographic Information*. 2020. URL: https://pgcc.libguides.com/c.php?g=60038&p=385730.

[11] James Connor. *Google Scholar Citations Open To All*. 2011. URL: https://scholar.googleblog.com/2011/11/google-scholar-citations-open-to-all.html.

[12] Raimo Streefkerk Courtney Gahan. *What is a DOI?* 2021. URL: https://www.scribbr.com/citing-sources/what-is-a-doi/.

[13] W. Cuthbertson. *What is a Scholarly Article and How Do I Find One*. 2020. URL: https://libguides.csuchico.edu/scholarly.

[14] W. Cuthbertson. *What is a Scholarly Article and How Do I Find One*. 2020. URL: https://libguides.csuchico.edu/scholarly#s-lib-ctab-9741173-1.

[15] DBLP. *DBLP dataset*. 2021. URL: https://dblp.uni-trier.de/xml/.

[16] DeepAI. *What is a Decision Tree in Machine Learning?* URL: https://www.ibm.com/cloud/learn/machine-learning.

[17] Google Developer. *What is Clustering?* 2020. URL: https://developers.google.com/machine-learning/clustering/overview.

[18] Scikit Learn Developers. *Choosing the right estimator*. 2021. URL: https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html.

[19] Scikit Learn Developers. *sklearn.model$_s$election.GridSearchCV*. 2021. URL: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html.

[20] Scikit Learn Developers. *sklearn.preprocessing.KBinsDiscretizer*. 2021. URL: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.KBinsDiscretizer.html.

[21] Cambridge Dictionary. *Meaning of cluster in English*. 2021. URL: https://developers.google.com/machine-learning/clustering/overview.

[22] Minitab Blog Editor. *Regression Analysis: How Do I Interpret R-squared and Assess the Goodness-of-Fit?* 2013. URL: https://blog.minitab.com/en/adventures-in-statistics-2/regression-analysis-how-do-i-interpret-r-squared-and-assess-the-goodness-of-fit.

[23] IBM cloud education. *MS Windows NT Kernel Description*. 2020. URL: https://www.ibm.com/cloud/learn/machine-learning.

[24] IBM Cloud Education. *Application Programming Interface (API)*. 2020. URL: https://www.ibm.com/cloud/learn/api.

[25] IBM Cloud Education. *Unsupervised Learning*. 2020. URL: https://www.ibm.com/cloud/learn/unsupervised-learning.

[26] L Egghe. *AN IMPROVEMENT OF THE H-INDEX: THE G-INDEX 1*. Tech. rep. 2006.

[27] Issam El Naqa and Martin J. Murphy. "What Is Machine Learning?" In: *Machine Learning in Radiation Oncology*. Springer International Publishing, 2015, pp. 3–11. DOI: 10.1007/978-3-319-18305-3_1.

[28] Ethem Alpaydin. *Introduction to Machine Learning*. 2014.

[29] Rodrigo Fernandes de Mello and Moacir Antonelli Ponti. *Machine Learning*. Cham: Springer International Publishing, 2018. ISBN: 978-3-319-94988-8. DOI: 10.1007/978-3-319-94989-5. URL: http://link.springer.com/10.1007/978-3-319-94989-5.

[30] Jason Fernando. *R-Squared Definition*. 2021. URL: https://www.investopedia.com/terms/r/r-squared.asp.

[31] Massimo Franceschet and Antonio Costantini. "The effect of scholar collaboration on impact and quality of academic papers". In: *Journal of Informetrics* 4.4 (2010), pp. 540–553. ISSN: 1751-1577. DOI: https://doi.org/10.1016/j.joi.2010.06.003. URL: https://www.sciencedirect.com/science/article/pii/S175115771000057X.

[32] Claudio Gambella, Bissan Ghaddar, and Joe Naoum-Sawaya. *Optimization problems for machine learning: A survey*. 2021. DOI: 10.1016/j.ejor.2020.08.045. arXiv: 1901.05331.

[33] Rohith Gandhi. *Introduction to Machine Learning Algorithms: Linear Regression*. 2018. URL: https://towardsdatascience.com/introduction-to-machine-learning-algorithms-linear-regression-14c4e325882a.

[34] Rohith Gandhi. *Support Vector Machine — Introduction to Machine Learning Algorithms*. 2018. URL: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47.

[35] Vicente P. Guerrero-Bote and Félix Moya-Anegón. "A further step forward in measuring journals' scientific prestige: The SJR2 indicator". In: *Journal of Informetrics* 6.4 (2012), pp. 674–688. ISSN: 17511577. DOI: 10.1016/j.joi.2012.07.001.

[36] Gongde Guo et al. “KNN model-based approach in classification”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 2888 (2003), pp. 986–996. ISSN: 16113349. DOI: 10.1007/978-3-540-39964-3_62.

[37] J E Hirsch. *An index to quantify an individual's scientific research output*. Tech. rep. 2005. URL: www.pnas.orgcgidoi10.1073pnas.0507655102.

[38] Eric Estevez Jake Frankenfield. *Artificial Neural Network (ANN)*. 2020. URL: https://www.investopedia.com/terms/a/artificial-neural-networks-ann.asp.

[39] javaTpoint. *Linear Regression in Machine Learning*. 2021. URL: https://www.javatpoint.com/linear-regression-in-machine-learning.

[40] M Jordan, J Kleinberg, and B Schölkopf. *Information Science and Statistics*. Tech. rep. 2006.

[41] Ashwini M. Joshi and Sameer Prabhune. “Random forest: A hybrid implementation for sarcasm detection in public opinion mining”. In: *International Journal of Innovative Technology and Exploring Engineering* 8.12 (2019), pp. 5022–5025. ISSN: 22783075. DOI: 10.35940/ijitee.L3758.1081219.

[42] Jasleen Kaur, Filippo Radicchi, and Filippo Menczer. “Universality of scholarly impact metrics”. In: *Journal of Informetrics* 7.4 (2013), pp. 924–932. ISSN: 17511577. DOI: 10.1016/j.joi.2013.09.002. arXiv: 1305.6339.

[43] Samiya Khan et al. “A survey on scholarly data: From big data perspective”. In: *Information Processing and Management* 53.4 (2017), pp. 923–944. ISSN: 03064573. DOI: 10.1016/j.ipm.2017.03.006.

[44] Kayvan Kousha and Mike Thelwall. “Can Google Scholar and Mendeley help to assess the scholarly impacts of dissertations?” In: *Journal of Informetrics* 13.2 (2019), pp. 467–484. ISSN: 1751-1577. DOI: https://doi.org/10.1016/j.joi.2019.02.009. URL: https://www.sciencedirect.com/science/article/pii/S1751157718304991.

[45] Wayne W. LaMorte. *The Multiple Linear Regression*. 2016. URL: https://sphweb.bumc.bu.edu/otlt/mph-modules/bs/bs704-ep713_multivariablemethods/bs704-ep713_multivariablemethods2.html.

[46] Scikit Learn. *scikit-learn*. 2021. URL: https://scikit-learn.org/stable/.

[47] Jiaying Liu et al. “A Survey of Scholarly Data Visualization”. In: *IEEE Access* 6 (2018), pp. 19205–19221. ISSN: 21693536. DOI: 10.1109/ACCESS.2018.2815030.

[48] Jahnavi Mahanta. *Introduction to Neural Networks, Advantages and Applications*. 2017. URL: https://towardsdatascience.com/introduction-to-neural-networks-advantages-and-applications-96851bd1a207.

[49] John Mccarthy. *WHAT IS ARTIFICIAL INTELLIGENCE?* Tech. rep. 2004. URL: http://www-formal.stanford.edu/jmc/.

[50] Lokman I. Meho and Kiduk Yang. “Impact of data sources on citation counts and rankings of LIS faculty: Web of science versus scopus and google scholar”. In: *Journal of the American Society for Information Science and Technology* 58.13 (2007), pp. 2105–2125. DOI: https://doi.org/10.1002/asi.20677. eprint: https://asistdl.onlinelibrary.wiley.com/doi/pdf/10.1002/asi.20677. URL: https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/asi.20677.

[51] Microsoft. *Open Academic Graph*. 2021. URL: https://www.microsoft.com/en-us/research/project/open-academic-graph/.

[52] Tom M. (Tom Michael) Mitchell. *Machine Learning*. 1997, p. 414. ISBN: 0070428077.

[53] Yubing Nie et al. “Academic rising star prediction via scholar's evaluation model and machine learning techniques”. In: *Scientometrics* 120.2 (2019), pp. 461–476.

[54] Ratna Astuti Nugrahaeni and Kusprasapta Mutijarsa. "Comparative Analysis of Machine Learning KNN, SVM, and Random Forests Algorithm for Facial Expression Classification". In: (2016).

[55] Pranjal Pandey. *Data Preprocessing: Concepts*. 2019. URL: https://towardsdatascience.com/data-preprocessing-concepts-fa946d11c825#:~:text=In%20any%20Machine%20Learning%20process,easily%20interpreted%20by%20the%20algorithm..

[56] Holy Python. *Decision Tree Pros Cons*. URL: https://holypython.com/dt/decision-tree-pros-cons/.

[57] Holy Python. *Support Vector Machine Pros Cons*. URL: https://holypython.com/svm/support-vector-machine-pros-cons/.

[58] Pranoy Radhakrishnan. *What are Hyperparameters ? and How to tune the Hyperparameters in a Deep Neural Network?* 2017. URL: https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a.

[59] CORE Computing Research and Education. *Conference Portal*. 2021. URL: http://portal.core.edu.au/conf-ranks/?search=&by=all&source=CORE2021&sort=atitle&page=1.

[60] Daniela Rosenstreich and Ben Wooliscroft. "Measuring the impact of accounting journals using Google Scholar and the g-index". In: *The British Accounting Review* 41.4 (2009), pp. 227–239. ISSN: 0890-8389. DOI: https://doi.org/10.1016/j.bar.2009.10.002. URL: https://www.sciencedirect.com/science/article/pii/S0890838909000614.

[61] Amiya Ranjan Rout. *ML – Advantages and Disadvantages of Linear Regression*. 2020. URL: https://www.geeksforgeeks.org/ml-advantages-and-disadvantages-of-linear-regression/.

[62] Stuart Russell and Peter Norvig. *Artificial Intelligence A Modern Approach*. SECOND EDI. 2003.

[63] Hassan Sayyadi and Lise Getoor. "Futurerank: Ranking scientific articles by predicting their future pagerank". In: *Proceedings of the 2009 SIAM International Conference on Data Mining*. SIAM. 2009, pp. 533–544.

[64] SCImago. *Scimago Journal Country Rank*. 2021. URL: https://www.scimagojr.com/journalrank.php.

[65] Muhammad Shafiq et al. "Data mining and machine learning methods for sustainable smart cities traffic classification: A survey". In: *Sustainable Cities and Society* 60 (2020). ISSN: 22106707. DOI: 10.1016/j.scs.2020.102177.

[66] Tarang Shah. *About Train, Validation and Test Sets in Machine Learning*. 2017. URL: https://www.ibm.com/cloud/learn/machine-learning.

[67] Neha Sharma, Reecha Sharma, and Neeru Jindal. "Machine Learning and Deep Learning Applications-A Vision". In: *Global Transitions Proceedings* 2.1 (2021), pp. 24–28. ISSN: 2666285X. DOI: 10.1016/j.gltp.2021.01.004.

[68] Arnab Sinha et al. "An overview of microsoft academic service (mas) and applications". In: *Proceedings of the 24th international conference on world wide web*. ACM. 2015, pp. 243–246.

[69] skLearn. *sklearn.tree.DecisionTreeClassifier*. 2021. URL: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html.

[70] Statistics Solutions. *What is Linear Regression*. 2013. URL: https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/what-is-linear-regression/.

[71] Amy M. Suiter and Heather Lea Moulaison. "Supporting Scholars: An Analysis of Academic Library Websites' Documentation on Metrics and Impact". In: *The Journal of Academic Librarianship* 41.6 (2015), pp. 814–820. ISSN: 0099-1333. DOI: https://doi.org/10.1016/j.acalib.2015.09.004. URL: https://www.sciencedirect.com/science/article/pii/S0099133315002062.

[72] Jie Tang, Duo Zhang, and Limin Yao. "Social Network Extraction of Academic Researchers". In: *ICDM'07*. 2007, pp. 292–301.

[73] Jie Tang et al. "A Combination Approach to Web User Profiling". In: *ACM TKDD* 5.1 (2010), pp. 1–44.

[74] Jie Tang et al. "A Unified Probabilistic Framework for Name Disambiguation in Digital Library". In: *IEEE Transactions on Knowledge and Data Engineering* 24.6 (2012), pp. 975–987.

[75] Jie Tang et al. "ArnetMiner: Extraction and Mining of Academic Social Networks". In: *KDD'08*. 2008, pp. 990–998.

[76] Jie Tang et al. "Topic Level Expertise Search over Heterogeneous Networks". In: *Machine Learning Journal* 82.2 (2011), pp. 211–237.

[77] FileInfo team. *.CSV File Extension*. 2021. URL: https://fileinfo.com/extension/csv.

[78] FileInfo team. *.JSON File Extension*. 2021. URL: https://fileinfo.com/extension/json.

[79] techopedia. *Artificial Neural Network (ANN)*. 2021. URL: https://www.techopedia.com/definition/5967/artificial-neural-network-ann.

[80] techopedia. *Training Data*. 2021. URL: https://www.techopedia.com/definition/33181/training-data.

[81] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning Data Mining, Inference, and Prediction*. 2001.

[82] Jack V Tu. *Advantages and Disadvantages of Using Artificial Neural Networks versus Logistic Regression for Predicting Medical Outcomes*. Tech. rep. 11. 1996, p. 12251231.

[83] Vatsal. *K-Nearest Neighbours Explained*. 2011. URL: https://towardsdatascience.com/k-nearest-neighbours-explained-7c49853633b6.

[84] Senzhang Wang et al. "Future influence ranking of scientific literature". In: *Proceedings of the 2014 SIAM International Conference on Data Mining*. SIAM. 2014, pp. 749–757.

[85] Feng Xia et al. "Big Scholarly Data: A Survey". In: *IEEE Transactions on Big Data* 3.1 (2017), pp. 18–35. ISSN: 2332-7790. DOI: 10.1109/tbdata.2016.2641460.

[86] Yinglin Xia. "Correlation and association analyses in microbiome study integrating multiomics in health and disease". In: *Progress in Molecular Biology and Translational Science*. Vol. 171. Elsevier B.V., 2020, pp. 309–491. DOI: 10.1016/bs.pmbts.2020.04.003.

[87] Ajay Yadav. *Decision Trees*. 2019. URL: https://towardsdatascience.com/decision-trees-d07e0f420175.

[88] Ying Ding, Ronald Rousseau, and Dietmar Wolfram. *Measuring Scholarly Impact Methods and Practice*. 2014, pp. 1–351. ISBN: 9783319103761.

[89] Tony Yiu. *Understanding Random Forest*. 2019. URL: https://towardsdatascience.com/understanding-random-forest-58381e0602d2.

[90] Wenhua Yu et al. “A Novel Practical Method to Classify Scholars Based on Research Subjects”. In: *2016 Intl IEEE Conferences on Ubiquitous Intelligence Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*. 2016, pp. 903–908. DOI: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0142.

[91] Zhaohui Wu et al. “Towards Building a Scholarly Big Data Platform:Challenges, Lessons and Opportunities”. In: (2014), p. 494.

[92] Zoila Ruiz, Jaime Salvador, and Jose Garcia-Rodriguez. “A Survey of Machine Learning Methodsfor Big Data”. In: Lecture Notes in Computer Science (2017). Ed. by José Manuel Ferrández Vicente et al. DOI: 10.1007/978-3-319-59773-7. URL: http://link.springer.com/10.1007/978-3-319-59773-7.